What is claimed is:

1. A method of generating an input file using a meta language regarding graphics data compression, the method comprising:

making an extensible markup language (XML) schema that defines at least a compression node describing object data to be compressed, and parameters used to data compression;

making style sheets which support conversion of an input XML file into a file, which is to be input to a data compression encoder, based on the XML schema; and

generating a file, which is to be input to the data compression encoder, by parsing the input XML file based on the XML schema and the style sheets.

2. The method of claim 1, wherein the XML schema further comprises at least EncodingHints specifying the location of a file in which the object data to be compressed is stored.

3. The method of claim 1, wherein the parameters comprise at least one of a parameter for keyframe-based animation data regarding vertex coordinates of the object, a parameter for rotation information of the object, a parameter for position information of the object to be compressed, and a parameter for three-dimensional (3D) mesh information to be compressed.

4. The method of claim 2, wherein the parameters comprise at least one of a parameter for keyframe-based animation data regarding vertex coordinates of the object, a parameter for rotation information of the object, a parameter for position information of the object to be compressed, and a parameter for three-dimensional (3D) mesh information to be compressed.

5. A method of generating an input file using a meta language regarding graphics data compression, the method comprising:

making an XMT schema which defines a compression node, which defines object data to be compressed, parameters for data compression, and BitWrapperEncodingHints which at least specifies the location of a file in which the object data to be compressed is stored;

making an XMT2BIFS style sheet which supports conversion of an XMT input file into an scene file and an XMT2MUX style sheet which supports conversion of the XMT input file into a mux file, based on the XMT schema; and

generating the scene file and the mux file by parsing the input XMT file using the XMT schema and the XMT2BIFS and XMT2MUX style sheets.

6. The method of claim 5, wherein the compression node comprises:

a node field containing the object data to be compressed;

a buffer field which is not used together with an url field at the same time and temporarily stores a bitstream defined in the compression node using an in-band scenario; and

the url field which is not used together with the buffer field at the same time and links information regarding the bitstream defined in the compression node using an out-band scenario.

7. The method of claim 6, wherein the compression node further comprises a type field specifying the type of node compression scheme.

8. The method of claim 5, wherein the parameters comprise at least one of a parameter for keyframe-based animation data regarding vertex coordinates of the object, a parameter for rotation information of the object, a parameter for position information of the object to be compressed, and a parameter for three-dimensional (3D) mesh information to be compressed.

9. The method of claim 5, wherein the BitWrapperEncodingHints further specifies an object descriptor ID that is the same as a URL ID, of the compression node, the name of a file transmitting a compressed bitstream, and the type of a stream format, the file name being described in the mux file.

10. The method of claim 5, wherein parsing the input XMT file further comprises:

receiving the input XMT file describing the compression node that defines the original data, compression parameters, and buffer; and generating the scene file and the mux file by parsing the input XMT file using the XMT schema and the XMT2BIFS and XMT2MUX style sheets,

wherein the scene file comprises the original data, compression parameters, and buffer which temporarily stores a bitstream obtained from compression of the original data, and

the mux file describes the name of a file obtained by encoding the scene file using a BIFS encoder and a stream format.

11. The method of claim 5, wherein parsing the input XMT file further comprises:

receiving the input XMT file defining the compression node containing the buffer temporarily storing the compressed object data; and

generating the scene file and the mux file by parsing the input XMT file using the XMT schema and the XMT2BIFS and XMT2MUX style sheets,

wherein the scene file contains the buffer temporarily storing the bitstream that is representation of the compressed object data, and

the mux file specifies the name of a file obtained by encoding the scene file using the BIFS encoder, and the stream format.

12. The method of claim 5, wherein parsing the input XMT file comprises:

receiving the input XMT file which describes the compression node defining the original data, compression parameters, and url information, and the BitWrapperEncodingHints defining object descriptor ID that is the same as an url ID the compression node and the location of the bitstream that is representation of the compressed object data; and

generating the scene file and the mux file by parsing the input XMT file using the XMT schema and the XMT2BIFS and XMT2MUX style sheets,

wherein the scene file contains the original data, compression parameters, and url information which links information regarding the bitstream obtained from compression of the original data, and

the mux file specifies the location of the bitstream, which is representation of the compressed object data, and the stream format defined in the BitWrapperEncodingHints.

13. The method of claim 12, wherein the input XMT file further comprises an ObjectDescriptorUpdate which defines an object descriptor ID that is the same as the object descriptor ID specified in the BitWrapperEncodingHints, and the name of a mux file to be generated from the parsing of the input XMT file,

wherein the scene file further specifies an object descriptor ID that is the same as the object descriptor ID specified in the BitWrapperEncodingHints, and the name of the mux file.

14. The method of claim 5, wherein parsing the input XMT file further comprises:

receiving the input XMT file describing the BitWrapperEncodingHints specifying the compression node defining the url which links information regarding already-compressed object data, an object descriptor ID that is the same as the url ID, and the location of the bitstream that is representation of the compressed object data; and

generating the scene file and the mux file by parsing the input XMT file using the XMT schema and the XMT2BIFS and XMT2MUX style sheets,

wherein the scene file containing the URL which links information regarding the bitstream obtained from the compression of the original data and whose ID is the same as the ID of the object descriptor specified in the compression node, and

the mux file specifies the location of the bitstream that is representation of the compressed object data defined in the BitWrapperEncodingHints and the stream format.

15. The method of claim 14, wherein the input XMT file further comprises an ObjectDescriptorUpdate which specifies an object descriptor ID that is the same as the object descriptor ID defined in the BitWrapperEncodingHints and the name of the mux file obtained by the parsing of the input XMT file,

wherein the scene file further specifies an object descriptor ID that is the same as the object descriptor ID defined in the BitWrapperEncodingHints and the name of the mux file.

16. A computer readable recording medium for recording a program executing the method of claim 1 in a computer.

17. A computer readable recording medium for recording a program executing the method of claim 5 in a computer.

18. A system for generating an input file using a meta language regarding graphics data compression, the system comprising:

an XML schema that defines a compression node at least containing information regarding object data to be compressed and parameters used for data compression;

style sheets which support conversion of the input XML file into a file which is to be input to a predetermined data compression encoder, based on the XML schema; and

an XML parser which parses the input XML file based on the XML schema and the style sheets to generate a file input to a predetermined data compression encoder.

19. The system of claim 18, wherein the parameters comprise at least one of a parameter for keyframe-based animation data regarding vertex coordinates of the object, a parameter for rotation information of the object, a parameter for position information of the object, and a parameter for three-dimensional (3D) mesh information to be compressed.

20. A system for generating an input file using a meta language regarding graphics data compression, the system comprising:

an XMT schema which defines a compression node specifying object data to be compressed, parameters for data compression, and a BitWrapperEncodingHints which at least specifies the location of a file in which the object data to be compressed is stored;

an XMT2BIFS style sheet which supports conversion of an input XMT file into a scene file based on the XMT schema;

an XMT2MUX style sheet which supports conversion of the input XMT file into a mux file based on the XMT schema; and

an XMT parser which parses the input XMT file based on the XMT schema and the XMT2BIFS and XMT2MUX style sheets to generate the scene and mux files as the input files to a predetermined compression encoder.

21. The system of claim 20, wherein the compression node comprises:

a node field specifying the object data to be compressed;

a buffer field which is not used together with a url field at the same time and transmits a bitstream that is representation of the compressed object data using the in-band scenario, the bitstream specified in the compression node; and

the url field which is not used together with the buffer field at the same time and transmits the bitstream that is representation of the compressed object data using the out-band scenario, the bitstream being specified in the compression node.

22. The system of claim 20, wherein the parameters comprise at least one of a parameter for keyframe-based animation data regarding vertex coordinates of the object, a parameter for rotation information of the object, a parameter for position information of the object, and a parameter for three-dimensional (3D) mesh information to be compressed.

23. The system of claim 20, wherein BitWrapperEncodingHints further comprises an object descriptor ID that is the same as an URL ID defined in the compression node, the name of a file transmitting the compressed bitstream contained in the mux file, and the type of the stream format.